IN THE CLAIMS

1. (Currently Amended) A method of extending EJB Handles for use with Enterprise JavaBeans (EJBs) to provide a SmartHandle, comprising the steps of:

maintaining an Entity EJB object relationship through a combination of a proxy pattern, an EJB Handle, and a primary key of the EJB Handle;

storing EJB Home class from which an Entity EJB was generated and from which said Entity EJB can be re-instantiated; ~~and~~

maintaining an instance of a SmartKey that describes said primary key for a database column to which an Entity EJB object is mapped; and

delegating to a SmartKey class that implements a java code to perform a field-by-field comparison of attributes associated with said primary key, thereby permitting two EJB Handles to be compared without instantiating the corresponding Entity EJB.

2. (Original) The method of claim 1, further comprising the steps of instantiating said Entity EJB object associated with said SmartHandle with a single method invocation.

3. (Original) The method of claim 2, wherein said single method invocation includes the steps of:

locating said EJB Home class using Java Naming and Directory Interface (JNDI);

using reflection to obtain an ejbFindByPrimaryKey method; and

invoking said ejbFindByPrimaryKey method with said SmartKey.

4. (Canceled)

5. (Original) The method of claim 1, wherein said SmartKey includes said primary key of the EJB Handle, thereby providing portability to said Entity EJB object.

It is requested the rejection of claims 3-4 and 10-11 under 35 U.S.C. §103 be withdrawn.

In addition, the Examiner alleges that Apte et al. teaches the steps of using reflection to obtain an ejbFindByPrimaryKey method and invoking the ejbFindByPrimaryKey method with the SmartKey, but dos not teach the step of loading the EJB Home class using JNDI, as required in claim 3 (and similarly in claim 10). To cure this deficiency in Apte et al., the Examiner turns to Acker et al. However, contrary to the Examiner's assertion, Apte et al. does not even teach or suggest the steps of using reflection and invoking the ejbFindByPrimaryKey recited in claim 4 (and similarly in claim 10). In fact, col. 7, lines 31-38 and col. 6, lines 58-67 in Apte et al., cited by the Examiner, merely describes that EJB can be invoked by clients or EJBs residing in one machine can be remotely invoked from another machine. Additionally, Apte et al. clearly states that "one cannot persist EJBs by storing their Home Name and Key Value" (col. 2, lines 36-37). Further, contrary to the Examiner's assertion, Acker et al. does not teach or suggest locating the EJB Home class using JNDI, as required in claim 3. In fact, paragraph [0006] in Acker et al., cited by the Examiner, merely describes that JNDI can be used to find object in the name space. Therefore, the combination of Apte et al. and Acker et al. does not teach or suggest the steps recited in claim 3 (and similarly in claim 10).

Statements appearing above in respect to the disclosures in the cited references represent the present opinions of the applicants' undersigned attorney and, in the event that the Examiner disagrees with any of such opinions, it is respectfully requested that the Examiner specifically indicate those portions of the reference providing the basis for a contrary view.

In view of the above, each of the presently pending claims in this application is believed to be in immediate condition for allowance. Accordingly, the Examiner is respectfully requested to pass this application to issue.

* * *

6. (Original) The method of claim 1, further comprising the step of assigning each attribute of said Entity EJB object and said SmartKey to a separate column within a relational database table, thereby permitting said SmartHandle to be mapped to a multi-column relational database table.

7. (Original) The method of claim 1, wherein said SmartHandle includes at least attributes HomeClassName, KeyClassName, and HomeName.

8. (Currently Amended) A SmartHandle for extending EJB Handles for use with Enterprise JavaBeans (EJBs), comprising:

EJB Home class from which an Entity EJB was generated and from which said Entity EJB can be re-instantiated; and

an instance of a SmartKey that describes said primary key for a database column to which an Entity EJB object is mapped; and

wherein said SmartHandle maintains an Entity EJB object relationship through a combination of a proxy pattern, an EJB Handle, and a primary key of the EJB Handle; and delegates to a SmartKey class that implements a java code to perform a field-by-field comparison of attributes associated with said primary key, thereby permitting two EJB Handles to be compared without instantiating the corresponding Entity EJB Objects.

9. (Original) The SmartHandle of claim 8, wherein said Entity EJB object associated with said SmartHandle being instantiated with a single method invocation.

10. (Original) The SmartHandle of claim 9, wherein said single method invocation is operable to locate said EJB Home class using Java Naming and Directory Interface (JNDI), use reflection to obtain an ejbFindByPrimaryKey method, and invoke said ejbFindByPrimaryKey method with said SmartKey.

11. (Canceled)

12. (Original) The SmartHandle of claim 8, wherein said SmartKey includes said primary key of the EJB Handle, thereby providing portability to said Entity EJB object.

13. (Original) The SmartHandle of claim 8, wherein each attribute of said Entity EJB object and said SmartKey is assigned to a separate column within a relational database table, thereby permitting said SmartHandle to be mapped to a multi-column relational database table.

14. (Original) The SmartHandle of claim 8, wherein said SmartHandle includes at least attributes HomeClassName, KeyClassName, and HomeName.